

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A) 平4-78284

⑬ Int. Cl.⁵H 04 N 7/01
5/253

識別記号

C

庁内整理番号

8838-5C
8942-5C

⑭ 公開 平成4年(1992)3月12日

審査請求 未請求 請求項の数 1 (全9頁)

⑮ 発明の名称 フレームレート変換装置

⑯ 特 願 平2-189998

⑰ 出 願 平2(1990)7月17日

⑱ 発 明 者 山 田 和 也 神奈川県横浜市神奈川区守屋町3丁目12番地 日本ビクター株式会社内

⑲ 出 願 人 日本ビクター株式会社 神奈川県横浜市神奈川区守屋町3丁目12番地

⑳ 代 理 人 弁理士 稲本 義雄

明 細 書

1. 発明の名称

フレームレート変換装置

2. 特許請求の範囲

入力画像データから距離 m ($m \geq 2$) のコアフレーム間の動きベクトルを検出し、前記動きベクトルのモードを渡すモードデータと、前記画像データとともに出力する符号器と、

前記符号器より入力される前記画像データを前記動きベクトルを用いてフレーム数 Q からフレーム数 P に変換する復号器とを備えるフレームレート変換装置において、

前記符号器は、前記コアフレーム間の動きベクトルを全て符号化して前記復号器に伝送するとともに、

前記復号器は、

前記符号器より伝送される前記動きベクトルを記憶する動きベクトルメモリと、

前記動きベクトルのブロックに対応するアドレスを発生するブロックアドレス発生器と、

前記フレーム数 Q と P により定まる係数を前記動きベクトルに乗算する乗算器と、

少なくとも連続する2つのフレームの画像データを記憶するフレームメモリと、

前記乗算器により補正された前記動きベクトルに対応する画像データを、前記フレーム数 Q と P により定まる係数で重み付けして、平均化する平均化回路とを有することを特徴とするフレームレート変換装置。

3. 発明の詳細な説明

〔産業上の利用分野〕

本発明はNTSC方式のフレーム数をPAL方式のフレーム数に変換したり、映画フィルムのコマ数をNTSC方式のフレーム数に変換する場合に用いて好適なフレームレート変換装置に関する。

〔従来の技術〕

第7図は、従来の画像データ処理装置の一例の構成を表わしている。

この装置は基本的に、符号器1と復号器2により構成されている。符号器1はフレームメモリ3、

特開平4-78284 (2)

4、動き検出回路5、動き補償回路6、動きベクトル選択回路7、マルチプレクサ(MPX)8および符号化回路9により構成されている。また、復号器2は、復号化回路11、ディマルチプレクサ(DMPX)12、フレームメモリ13乃至15、動き補償回路16およびマルチプレクサ(MPX)17により構成されている。

図示せぬ回路より供給された画像データはフレーム毎にフレームメモリ3と4に蓄えられる。動き検出回路5はフレームメモリ3と4に蓄えられた画像データから所定の範囲(例えば $n \times n$ 画素)の画像データを抜き出し、動きの方向と量(動きベクトル)を検出する。動き補償回路6は動きベクトルに対応して画像データを補償し、補償した画像データをマルチプレクサ8に出力する。また、このとき動き補償を行ったモードもマルチプレクサ8に供給される。さらに、動き補償を行う上で必要であった動きベクトルが動きベクトル選択回路7により選択され、マルチプレクサ8に供給される。

を加算し、平均化したものに近似しているとき、両方のフレームからの動きベクトルに対応して画像データが生成される。これらのうち、いずれの動き補償が行われたのかは、モードデータとして符号化される。

復号化回路11は符号化回路9より入力されたデータを復号化し、ディマルチプレクサ12に出力する。ディマルチプレクサ12は入力されたデータをディフォーマットし、画像データ、モードデータ、MVデータに分離する。画像データはフレームメモリ13、14に順次蓄えられる。モードデータとMVデータは動き補償回路16に入力される。動き補償回路16はモードデータとMVデータに対応するブロックの画像データをフレームメモリ13、14から読み出し、動き補償する。この動き補償により得られた画像データはフレームメモリ15に蓄えられる。マルチプレクサ17はフレームメモリ14又は15より読み出された画像データを選択し、出力する。

〔発明が解決しようとする課題〕

マルチプレクサ8は、入力された画像データ、モードデータおよび動きベクトル(MV)データを所定のフォーマットにまとめ、符号化回路9に出力する。符号化回路9は入力されたデータを所定の規則に従って符号化し、復号化回路11に伝送する。

次に、第8図を参照して動き補償の動作について説明する。

いま、例えばフレーム間距離 m が3のフレーム0, 3, 6, ...をコアフレームとすると、これらのフレームのデータはフレーム内処理で符号化される。これに対して、インターフレーム1, 2(4, 5)はコアフレーム0, 3(3, 6)を動き補償して符号化される。

例えばフレーム1は、コアフレーム0またはコアフレーム3に最も近似しているとき、これらのフレームの動きベクトルから動き補償が行われる(画像データが生成される)。また、コアフレーム0と3のいずれにも近似していないときはフレーム内処理される。さらに、コアフレーム0と3

しかしながら、この従来の装置を、例えば1秒間に24コマ(フレーム)のフィルム画像を、1秒間に30フレーム(60フィールド)のNTSC方式の画像に変換するフレームレート変換装置に適用しようとするとき、伝送されるMVデータは符号化時に必要なものだけであるため、マルチプレクサ17より出力される画像データから新たに動きベクトルを検出しなければならず、構成が複雑になり、不利であった。

本発明はこのような状況に鑑みてなされたもので、簡単な構成でフレームレートを変換できるようにするものである。

〔課題を解決するための手段〕

本発明のフレームレート変換装置は、入力画像データから距離 m ($m \geq 2$)のコアフレーム間の動きベクトルを検出し、動きベクトルのモードを表すモードデータと、画像データとともに出力する符号器と、符号器より入力される画像データを動きベクトルを用いてフレーム数 Q からフレーム数 P に変換する復号器とを備えるフレームレート

特開平4-78284 (3)

変換装置において、符号器は、コアフレーム間の動きベクトルを全て符号化して復号器に伝送するとともに、復号器は、復号器より伝送される動きベクトルを記憶する動きベクトルメモリと、動きベクトルのブロックに対応するアドレスを発生するブロックアドレス発生器と、フレーム数 Q と P により定まる係数を動きベクトルに乗算する乗算器と、少なくとも連続する2つのフレームの画像データを記憶するフレームメモリと、乗算器により補正された動きベクトルに対応する画像データを、フレーム数 Q と P により定まる係数で重み付けして、平均化する平均化回路とを有することを特徴とする。

〔作用〕

上記構成のフレームレート変換装置においては、コアフレーム間の動きベクトルが全て復号器側に伝送される。そして、変換前後のフレーム数に対応して定まる係数で動きベクトルが重み付けされ、平均化される。従って、簡単な構成でフレームレートの変換が可能になる。

n 画素の動きベクトルのブロックに対応するアドレスを発生し、動きベクトルメモリ25に供給するとともに、加算器32、33を介してフレームメモリ23、24に供給する。

動きベクトルメモリ25は、フレームメモリ23と24に記憶されたフレームの直前および直後のコアフレーム間の動きベクトルを乗算器26に出力する。乗算器26は入力された動きベクトルに係数 $1/m$ (m はコアフレーム間距離)を乗算し、乗算結果を後段の乗算器28に出力する。乗算器28はフレーム数カウンタ40が出力するカウンタ値に対応して係数 K 。乃至 K_{P-1} を巡回させ、そのいずれかを選択する。そして選択した係数を入力された動きベクトルデータに乗算する。乗算器28の出力はさらに乗算器30に入力され、係数 $1/P$ が乗算される。この値 P は、出力される画像のフレーム数(変換後のフレーム数)に対応している。乗算器30の出力が加算器32に入力され、ブロックアドレス発生器34から入力されるブロックアドレスに加算され、フレームメモ

〔実施例〕

第1図は本発明のフレームレート変換装置の一実施例の構成を示すブロック図である。

符号器21と復号器22は、基本的に第7図に示した符号器1と復号器2に対応している。但し符号器21には第7図における動きベクトル選択回路7が設けられておらず、符号化時に必要な動きベクトルだけでなく、全ての動きベクトル(MV)データが復号器22に伝送されるようになっている。

復号器22は、符号器21より入力される動きベクトルとモードデータを用いて、やはり符号器21より入力される画像データを動き補償し、動き補償した画像データをフレームメモリ23、24に順次出力し、蓄込させる。すなわち、このフレームメモリ23、24には、連続する2つの静止画フレームの画像データが記憶される。

一方、復号器22より出力された動きベクトルデータは動きベクトルメモリ25に入力され、記憶される。ブロックアドレス発生器34は、 $n \times$

リ23に出力される。

この乗算器26、28、30と同様に、フレームメモリ24に記憶されたフレームのデータを処理するため、乗算器27、29、31が設けられている。

フレームメモリ23より読出された画像データは乗算器35に入力され、所定の係数 K_{P-1} 乃至 K のいずれかが乗算される。これらの係数は、乗算器28における場合と同様に、フレーム数カウンタ40のカウント値に対応して巡回するようになっている。乗算器35の出力は加算器37に入力されている。

乗算器35に対応して乗算器36が設けられており、フレームメモリ24の出力に所定の係数 K 乃至 K_{P-1} のいずれかが乗算された後、加算器37に出力されるようになっている。加算器37は乗算器35、36の出力を加算し、乗算器38に出力している。乗算器38は入力データに係数 $1/P$ を乗算し、フレームメモリ39に出力する。フレームメモリ39とフレーム数カウンタ4

特開平4-78284 (4)

0にはフレーム番号が入力されている。

なお、乗算器35、36、38と加算器37は平均化回路を構成している。

コアフレーム間距離 m を3、変換前のフレーム数 Q を6、変換後のフレーム数 P を5とするとき、第1図の回路の主要部は、第2図に示ようになる。すなわち、乗算器26と27における係数 $\pm 1/m$ は、それぞれ $\pm 1/3$ となり、乗算器30、31、38の係数 $1/P$ は $1/5$ となる。また、乗算器28と36の係数 K 乃至 K_{p-1} は、0乃至4、乗算器29と35の係数 K_{p-1} 乃至 K は5乃至1となる。

次に、第3図を参照して、第2図の実施例の動作を説明する。

いまコアフレーム間距離 m が3であるから、入力画像のフレーム0、3、6がコアフレームとされる。例えばフレームメモリ23にフレーム1の画像データが、また、フレーム24にフレーム2の画像データが、それぞれ記憶されているとすると、そのタイミングにおいて、動きベクトルメモ

リ25には、フレーム1とフレーム2の直前と直後のコアフレームであるフレーム0とフレーム3の間の動きベクトル $MV0$ が記憶されている。この動きベクトル $MV0$ は乗算器26に入力され、係数 $1/m (= 1/3)$ が乗算される。これにより、フレーム1とフレーム2の間の動きベクトルが求められる。フレーム0とフレームAのタイミングが一致しているものとする、入力画像データのフレーム(フレーム1乃至6)に対して出力画像データのフレーム(フレームA乃至F)は、1フレーム間距離の $1/5$ ずつずれていく。そして、このずれ量は0、 $1/5$ 、 $2/5$ 、 $3/5$ 、 $4/5$ 、0、 $1/5$ 、…と順次変化する。フレームBはフレーム1に対して動きベクトル $MV0$ の方向に $1/5$ だけずれている。そこで、乗算器28において係数1が選択される。これにより、乗算器30の係数 $1/5$ と合わせて、動きベクトル $(1/3)MV0$ に係数 $1/5$ が乗算され、

$$(1/3) \times MV0 \times (1/5)$$

が得られることになる。また、このとき、ブロッ

クアドレス発生器34は、動きベクトル $(1/3)MV0$ に対応するフレーム1のブロックのアドレス $F(1)$ を出力しているの、加算器32において、次の演算が行なわれる。

$$F(1) + (MV0 \times 1) / (3 \times 5)$$

加算器32より上式で示すアドレスが入力されるので、フレームメモリ23より、このアドレスで指定する画像データ、

$$IMD[F(1) + (MV0 \times 1) / (3 \times 5)]$$

が読出される。

一方、第3図に示すように、フレームBは、フレーム2から、動きベクトル $(1/3)MV0$ と反対方向に、フレーム間距離の $4/5$ だけずれているから、そのずれ量は次式で示すことができる。

$$F(2) - (MV0 \times 4) / (3 \times 5)$$

上述した場合と同様に、乗算器27、29、31および加算器33が上式の演算を実行する。これにより、フレームメモリ24からこのアドレスで指定するデータ、

$$IMD[F(2) - (MV0 \times 4) / (3 \times 5)]$$

が読出される。

フレームBの画像はフレーム1とフレーム2の画像を合成して得られるが、フレーム1とフレーム2のフレームBに与える影響は、フレーム1とフレームBの距離とフレーム2とフレームBの距離に逆比例するものと考えられる。すなわち、フレームBの画像データ $FD(B)$ は、次式を演算することにより求められる。

$$FD(B) = (4/5)IMD[F(1) + (MV0 \times 1) / (3 \times 5)] + (1/5)IMD[F(2) - (MV0 \times 4) / (3 \times 5)]$$

上式の演算を行うため、フレームメモリ23より読出されたデータに乗算器35で係数4が乗算され、フレームメモリ24より読出されたデータに乗算器36で係数1が乗算される。そして、乗算器35と36の出力が加算器37で加算された後、乗算器38で係数 $1/5$ が乗算される。このようにして、得られた画像データがフレームメモリ39に書込まれる。

フレーム数カウンタ40は入力されるフレーム信号をカウントし、カウント値に対応して乗算器

特開平4-78284 (5)

28, 29, 35, 36の係数を巡環させる。このようにして、次式で示すようにフレームA乃至Eの画像データFD(A)乃至FD(E)がフレームメモリ39に書き込まれる。

$$\begin{aligned} FD(A) &= (5/5) \text{IND}[F(0) + (MV0 \times 0) / (3 \times 5)] \\ &\quad + (0/5) \text{IND}[F(1) - (MV0 \times 5) / (3 \times 5)] \\ FD(B) &= (4/5) \text{IND}[F(1) + (MV0 \times 1) / (3 \times 5)] \\ &\quad + (1/5) \text{IND}[F(2) - (MV0 \times 4) / (3 \times 5)] \\ FD(C) &= (3/5) \text{IND}[F(2) + (MV0 \times 2) / (3 \times 5)] \\ &\quad + (2/5) \text{IND}[F(3) - (MV0 \times 3) / (3 \times 5)] \\ FD(D) &= (2/5) \text{IND}[F(3) + (MV1 \times 3) / (3 \times 5)] \\ &\quad + (3/5) \text{IND}[F(4) - (MV1 \times 2) / (3 \times 5)] \\ FD(E) &= (1/5) \text{IND}[F(4) + (MV1 \times 4) / (3 \times 5)] \\ &\quad + (4/5) \text{IND}[F(5) - (MV1 \times 1) / (3 \times 5)] \end{aligned}$$

第4図は他の実施例の構成を示している。この実施例においては、第1図(第2図)における動きベクトルメモリ25が、動きベクトルメモリ51と53、補間処理回路52により構成されている。そして、第1図(第2図)の実施例においてはブロックアドレス発生器34が発生するブロッ

クアドレスが、動きベクトル処理単位の $n \times n$ 画素のブロックアドレスであるのに対し、この実施例のブロックアドレス発生器54が発生するブロックアドレスは、 $r \times r$ ($r < n$)画素のブロックアドレスとされている。その他の構成は第1図(第2図)における場合と同様である。

$n \times n$ 画素のブロックA乃至Dと、 $r \times r$ 画素のブロックa乃至dの関係は例えば第5図に示すようになっている。

すなわちこの場合、 $r = n / 2$ とされている。そして、 $n \times n$ 画素のブロックAの右にブロックB、下にブロックC、右下にブロックDが存在するとき、ブロックAの中央に $r \times r$ 画素のブロックaが配置される。その結果、ブロックaの右側隣りのブロックbは、左半分がブロックAに、右半分がブロックBに属している。同様に、ブロックaの下に隣接するブロックcは、上半分がブロックAに、下半分がブロックCに属することになる。さらに、ブロックaの右下に隣接するブロックdは、それぞれ $1/4$ が、ブロックA乃至Dに

属することになる。

そこで、第4図の実施例の場合、復号器22より出力された動きベクトルが動きベクトルメモリ51に入力され、一旦記憶される。そして、ブロックA乃至Dの動きベクトルをそれぞれMVA, MVB, MVC, MVD、ブロックa乃至dの動きベクトルをそれぞれMVA, MVB, MVB, MVBとすると、補間処理回路52は動きベクトルメモリ51から動きベクトルMVA乃至MVDを読出し、次式より動きベクトルMVA乃至MVBを演算する。

$$\begin{aligned} MVA &= MVA \\ MVB &= (MVA + MVB) / 2 \\ MVC &= (MVA + MVC) / 2 \\ MVD &= (MVA + MVB + MVC + MVD) / 4 \end{aligned}$$

このようにして得られた $r \times r$ 画素の動きベクトルが上述した第1図(第2図)における場合と同様に処理される。

このように、補間するブロックのサイズを小さくすることにより、細かい動きに対しても対応が

可能になる。

なお、周辺の領域(例えばブロックaの上方の $r \times (r/2)$ の領域)の動きベクトルは、その内側のブロック(ブロックa)と同一とすることができる。

第6図はさらに他の実施例を示している。

この実施例においては乗算器35, 36の出力がそれぞれ乗算器62, 63を介して加算器37に供給されている。そして、この乗算器62, 63の係数が、モード情報メモリ61の出力に対応して制御されるようになっている。その他の構成は第1図(第2図)における場合と同様である。

復号器22が出力するモード情報はモード情報メモリ61に記憶される。第8図を参照して先に説明したようにモードには4種類ある。例えば第3図において、フレーム1をフレーム0と3から動き補償する場合、第1のモードにおいては、フレーム1は独立にフレーム内処理される。これはフレーム1がフレーム0と3とは全く異なる画像の場合である。第2のモードにおいては、フレ

特開平4-78284 (6)

ム1はフレーム0から、また、第3のモードにおいてはフレーム1はフレーム3から、それぞれ動き補償される。これは、フレーム1とフレーム3の間、またはフレーム0とフレーム1の間で、画像が全く切替わっているような場合である。さらに第4のモードにおいては、フレーム1はフレーム0とフレーム3の両方から動き補償される。これは、フレーム0からフレーム3にかけて、画像が関連している場合である。

このような4つのモードに対応して、第1表に示すように、乗算器62と63の係数が設定される。

第1表

モード	乗算器	
	62	63
1	1	1
2	P	0
3	0	P
4	1	1

ロック図、第3図は第2図の実施例の動作を説明するフレーム変換図、第4図および第6図は本発明のフレームレート変換装置の他の実施例の構成を示すブロック図、第5図は第4図の実施例の動作を説明するブロックの単位の図、第7図は従来の画像処理装置の一例の構成を示すブロック図、第8図は第7図の例の動作を説明する図である。

1, 21…符号器、2, 22…復号器、23, 24, 39…フレームメモリ、25, 51, 53…動きベクトルメモリ、26乃至31, 35, 36, 38…乗算器、34…ブロックアドレスカウンタ、40…フレーム数カウンタ。

特許出願人 日本ビクター株式会社

代理人 井理士 橋本 義雄

モード2のとき乗算器63の係数が0に設定され、乗算器62の係数がPに設定されるので、前のフレームからの動き補償のみが行なわれる。また、モード3のとき乗算器62の係数が0に設定され、乗算器63の係数がPに設定されるので、後のフレームからの動き補償のみが行なわれる。モード1および4のときは、乗算器62と63の両方の係数が1に設定されるので、第2図(第2図)における場合と同様の処理が行なわれる。

(発明の効果)

以上の如く本発明のフレームレート変換装置によれば、コアフレーム間の動きベクトルを全て復号器側に伝送し、変換前後のフレーム数に対応して定まる係数で動きベクトルを重み付けし、平均化するようにしたので、簡単な構成でフレームレートを変換することが可能になる。

4. 図面の簡単な説明

第1図は本発明のフレームレート変換装置の一実施例の構成を示すブロック図、第2図は第1図の実施例において所定の条件を設定した場合のブ

特開平4-78284 (7)

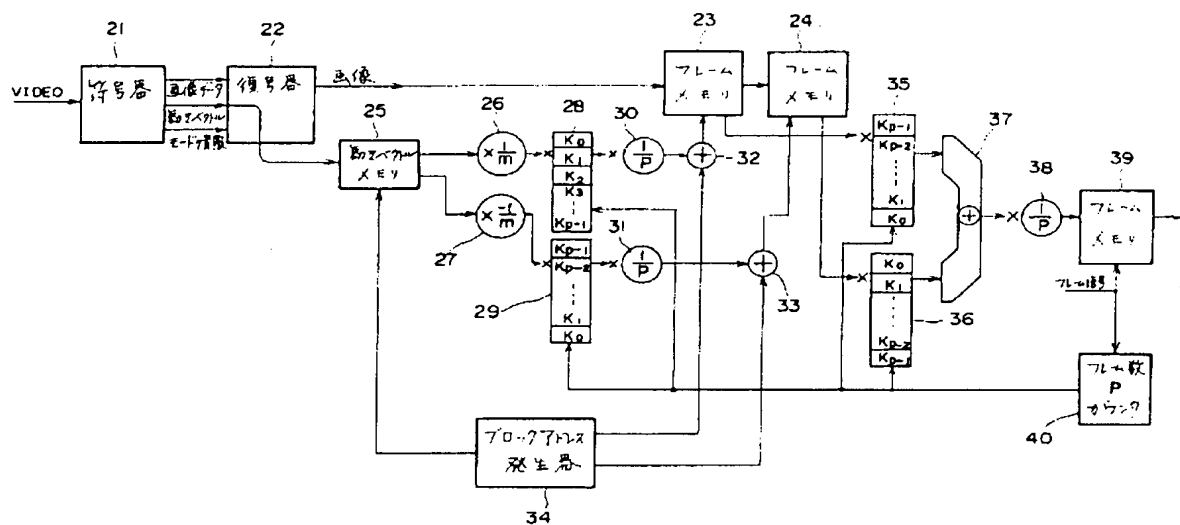


図 1

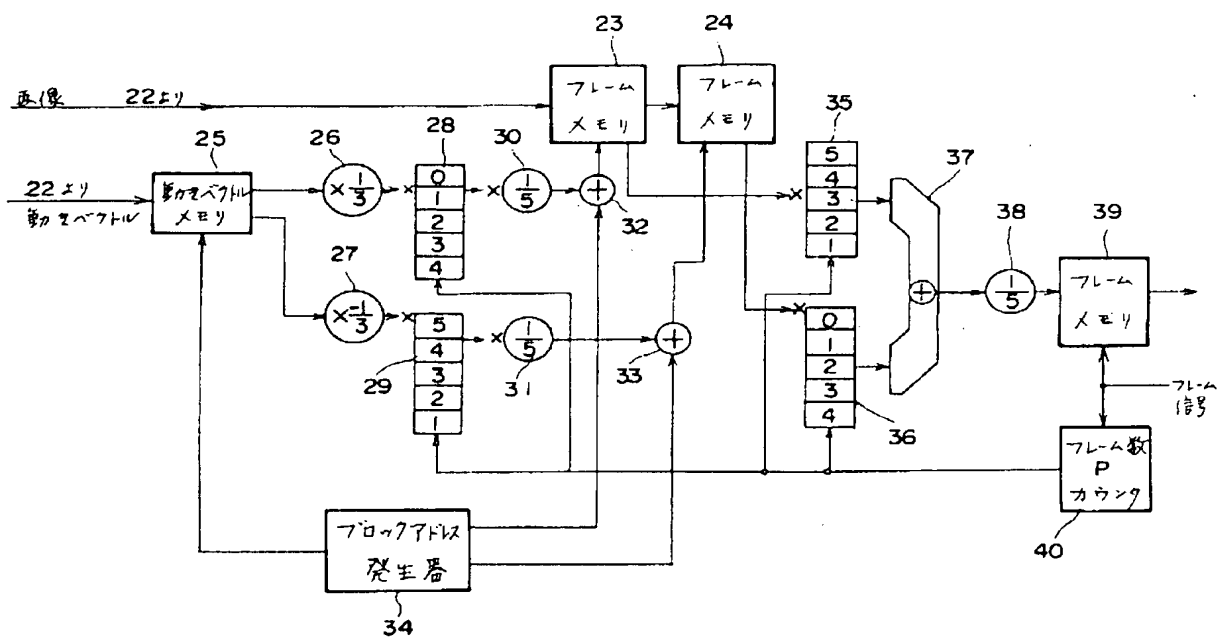
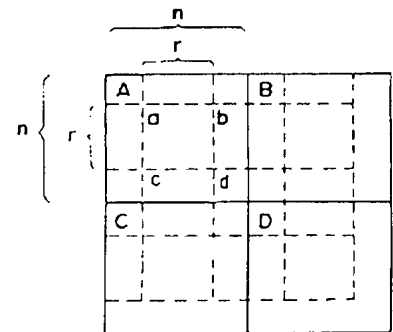
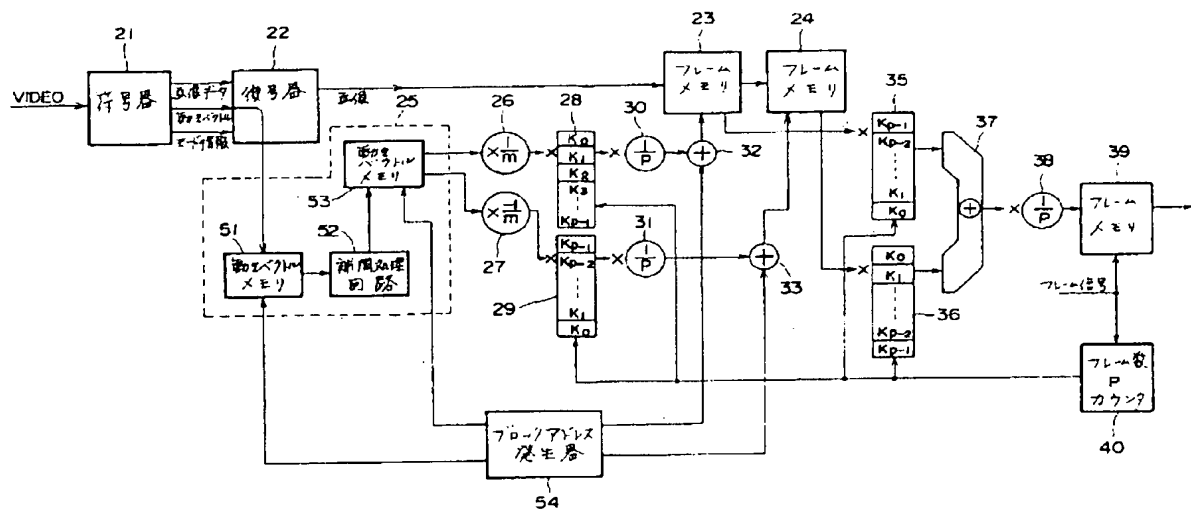


図 2

第 5 図

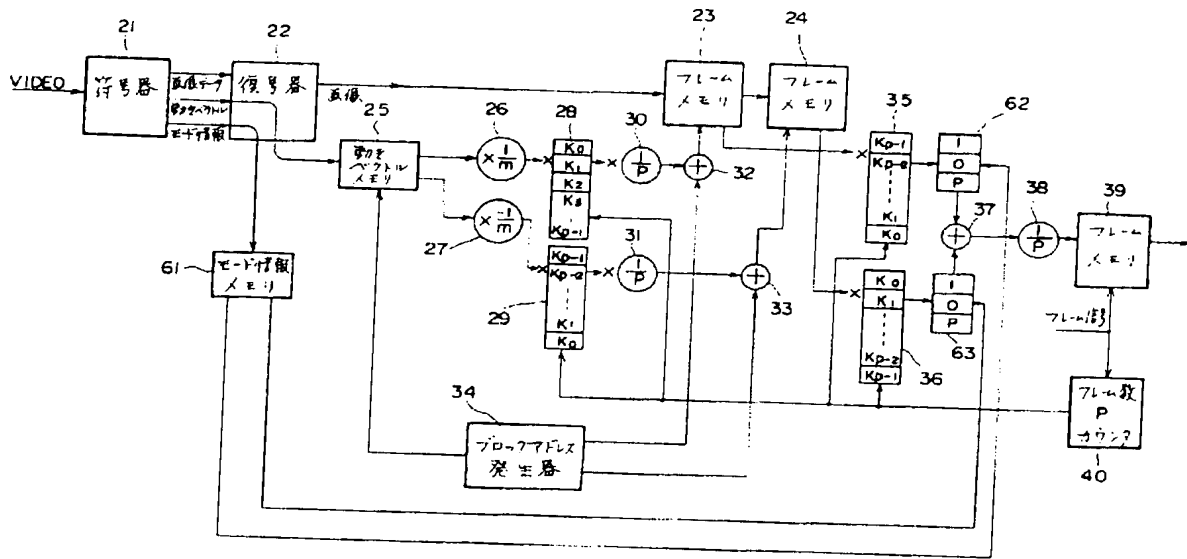


第 3 回

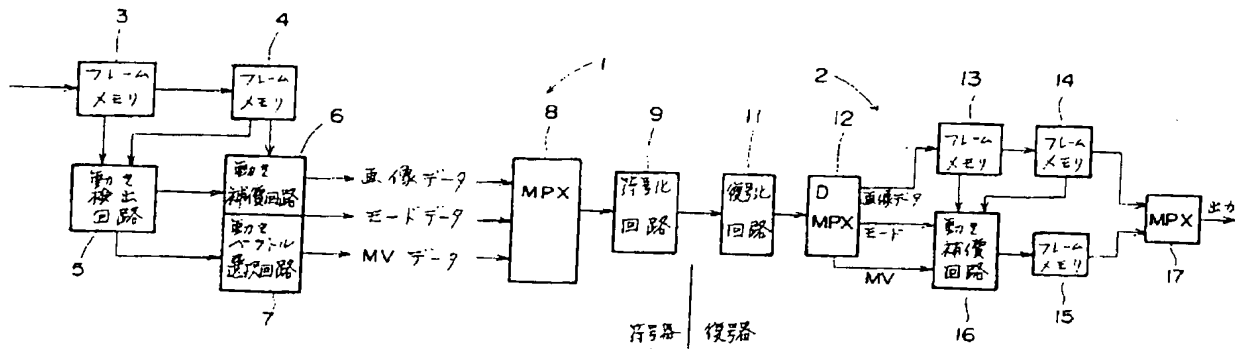


第 4 回

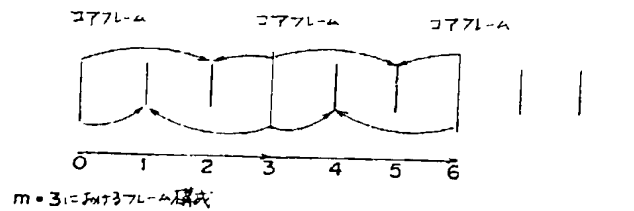
特開平4-78284 (9)



第 6 図



第 7 図



第 8 図

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 04-078284

(43)Date of publication of application : 12.03.1992

(51)Int.Cl. H04N 7/01

H04N 5/253

(21)Application number : 02-189998 (71)Applicant : VICTOR CO OF JAPAN
LTD

(22)Date of filing : 17.07.1990 (72)Inventor : YAMADA KAZUYA

(54) FRAME RATE CONVERTER

(57)Abstract:

PURPOSE: To simplify the frame rate conversion by weighting a moving vector with a coefficient depending on a frame number before and after the conversion and averaging the result of weighting.

CONSTITUTION: A coefficient is multiplied with a data read from a frame memory 23 at a multiplier 35 and a coefficient is multiplied with a data read from a frame memory 24 at a multiplier 36. Then after outputs of the multipliers 35,36 are added by an adder 37 and a coefficient $1/5$ is multiplied with the sum by a multiplier 38. The picture data obtained in this way is written in a frame memory 39. A frame number counter 40 counts inputted frame signals and circulates coefficients for the multipliers 28, 29, 35, 36 depending on the count and writes picture data FD(A)-FD(E) to a frame memory 39.

LEGAL STATUS [Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平4-78284

(43) 公開日 平成 4 年 (1992) 3 月 12 日

(51) Int. Cl. ⁵
H 0 4 N 7/01
5/253

識別記号

F I

審査請求 未請求 請求項の数 1 (全 9 頁) (11)

(21) 出願番号 特願平2-189998
(22) 出願日 平成 2 年 (1990) 7 月 17 日

(71) 出願人 000000432
日本ビクター株式会社
神奈川県
(72) 発明者 山田 和也
*

(54) 【発明の名称】フレームレート変換装置

(57) 【要約】

【目的】フレームレート変換装置に適用しようとする
と、伝送されるモードデータおよび動きベクトルデータ
は符号化時に必要なものだけである為、マルチプレクサ
より出力される画像データから新たに動きベクトルを検
出しなければならず、構成が複雑になるという課題を解
決する

【効果】コアフレーム間の動きベクトルを全て復号器側
に伝送し、変換前後のフレーム数に対応して定まる係数
で動きベクトルを重み付けし、平均化するようにしたの
で、簡単な構成でフレームレートを変換することが可能
になる

【産業上の利用分野】フレームレート変換装置に関する

【特許請求の範囲】

請求の範囲テキストはありません。

【発明の詳細な説明】

詳細な説明テキストはありません。

【図面の簡単な説明】

図面の簡単な説明テキストはありません。